TRANSMITTAL FORM	Attorney Docket No. SVL920010001US1 2061P

In re the application **Jason Alexander FCU, et al.**

Serial No: **09/820,451**

Filed: **March 28, 2001**

Confirmation No: **6671**

Group Art Unit: **2171**

Examiner: **Chen, Te Y.**

For: **METHOD AND SYSTEM FOR PROVIDING A GENERIC SCALAR FUNCTION**

ENCLOSURES *(check all that apply)*					
☐	Amendment/Reply	☐	Assignment and Recordation Cover Sheet	☐	After Allowance Communication to Group
	☐ After Final	☐	Part B-Issue Fee Transmittal	☐	Appeal Communication to Board of Appeals and Interferences
☐	Information disclosure statement	☐	Letter to Draftsman	■	Appeal Communication to Group *(Appeal Brief- in triplicate))*
	☐ Form 1449	☐	Drawings	☐	Status Letter
	☐ (X) Copies of References	☐	Petition	■	Postcard
☐	Extension of Time Request *	☐	Fee Address Indication Form	☐	Other Enclosure(s) *(please identify below):*
☐	Express Abandonment	☐	Terminal Disclaimer		
☐	Certified Copy of Priority Doc	☐	Power of Attorney and Revocation of Prior Powers		
☐	Response to Incomplete Appln	☐	Change of Correspondence Address		
☐	Response to Missing Parts	*Extension of Term: Pursuant to 37 CFR 1.136, Applicant petitions the Commissioner to extend the time for response for xxxxxx month(s), from to .			
	☐ Executed Declaration by Inventor(s)				

CLAIMS					
FOR	Claims Remaining After Amendment	Highest # of Claims Previously Paid For	Extra Claims	RATE	FEE
Total Claims	0	0	0	$18.00	$ 0.00
Independent Claims	0	0	0	$86.00	$ 0.00
				Total Fees	$ 0.00

METHOD OF PAYMENT	
☐	Check no. _____ in the amount of $ _____ is enclosed for payment of fees.
■	Charge $ 330.00 to Deposit Account No. 09-0460 (IBM Corporation) for payment of Appeal Brief filing fee.
■	Charge any additional fees or credit any overpayment to Deposit Account No. 09-0460 (IBM Corporation)

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT	
Attorney Name	Stephen G. Sullivan, Reg. No. 38,329
Signature	
Date	June 29, 2004

CERTIFICATE OF MAILING	
I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on this date: June 29, 2004	
Type or printed name	Jackie Tanda
Signature	Jackie Tanda

AF/2171

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

APPEAL NO:

In Re Application of: Date: June 29, 2004

Jason Alexander CU, et al. Confirmation No: 6671

Serial No: 09/820,451 Group Art Unit: 2171

Filed: March 28, 2001 Examiner: Chen, T.

For: METHOD AND SYSTEM FOR PROVIDING A GENERIC SCALAR FUNCTION

RECEIVED
JUL 0 8 2004
Technology Center 2100

APPELLANT'S BRIEF

Attorney for Appellants
INTERNATIONAL BUSINESS MACHINES
Sawyer Law Group LLP

TOPICAL INDEX

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

APPEAL NO:

In Re Application of: Jason Alexander CU, et al.

Serial No: 09/820,451

Filed: March 28, 2001

Date: June 29, 2004

Confirmation No: 6671

Group Art Unit: 2171

Examiner: Chen, T.

For: METHOD AND SYSTEM FOR PROVIDING A GENERIC SCALAR FUNCTION

Mail Stop Appeal Brief – Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPELLANT'S BRIEF ON APPEAL

Sir:

Appellant herein files an Appeal Brief drafted in accordance with the provisions of 37 C.F.R. § 1.192(c) as follows:

I. REAL PARTY IN INTEREST

Appellant respectfully submits that the above-captioned application is assigned, in its entirety to International Business Machines of Armonk, New York.

II. RELATED APPEALS AND INTERFERENCES

Appellant states that, upon information and belief, he is not aware of any co-pending appeal or interference which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. STATUS OF CLAIMS

Claims 1-21 are pending.

IV. STATUS OF AMENDMENT

The proposed Amendment to the specification in response to the Final Office Action was not entered.

V. SUMMARY OF THE INVENTION

The present invention provides a method, system and computer-readable media for utilizing a column function for a relational database in a structure query language (SQL) environment. In particular, the method, system and computer-readable media allow conventional, pre-existing column functions to operate on data other than columns.

Relational databases organize data into entries that are placed into one or more columns and one or more rows of a table. Such a relational database is typically used to archive information. Specification, page 1, lines 7-14. The column function is a conventional column function that is already in existence and is used in performing an operation on an indeterminate number of entries in a table of a relational database. More specification, a column function is a pre-existing function that typically operates on data that has been organized into one or more

columns. Specification, page 3, lines 1-3. The existence of and operation of the pre-existing column function is already known. Specification, page 2, lines 11-17 and page 3, lines 8-12 (giving as examples of column functions the minimum, maximum, sum and average functions).

In the method, system, and computer-readable medium in accordance with the present invention, a generalized scalar function is used to allow other row data to be operated on by a column function. At least one row of data is specified as an argument for the generalized scalar function. Specification, page 7, lines 21-23. The generalized scalar function is used to simulate a column environment for the row(s) that is the argument for the generalized scalar function. Specification, page 8, lines 10-13. For example, as indicated in Paragraph 7 of the Declaration under 37 C.F.R. 1.132 (Declaration), the entries in the row could simply be provided one-by-one to the column function. See also, specification, page 9, lines 15-18 and page 10, lines 1-3. Thus, the generalized scalar function allows data in the row(s) to be provided to the column function as though the row(s) were column(s). Specification, page 8, lines 15-17. The column function is performed on the data in the row(s) to provide the output(s) of the column function. Stated differently, the column function operates in a conventional manner on the row entries provided to the column function. Specification, page 8, lines 17-20. Thus, the generalized scalar function is used to provide the entries in a row to the column function as though the entries came from a column. Specification, page 8, lines 15-17. As a result, in addition to performing operations in a conventional manner on column data, the column function can perform its operations on row(s). Specification, page 8, lines 17-19. Consequently, column functions, such minimum, maximum, sum, and average can be performed on rows without requiring the column functions to be rewritten. Specification, page 8, lines 1-6. As a result, resources are conserved.

For example, Figure 4 describes one embodiment of a method that can be used for data in a table such at the table 1 depicted in Figure 1. In such an embodiment, a user specifies which row(s) is to be used for data. Specification, page 9, lines 12-13 and Figure 4, item 152. Stated differently, the row(s) are specified as the argument for the generalized scalar function. Referring to Figures 1 and 4, suppose that the user specifies that the row 6 of the table 1 is the row to be operated on. The generalized scalar function fetches the first entry (11) and provides the entry to the column function. Specification, page 9, lines 13-18 and Figure 4, items 154 and 156. The column function then operates on the entry normally. Specification, page 9, line 17-page 10, line 1 and Figure 4, items 158, 160, 166 and 168. It is determined whether the entry provided to the column function is the last in the row and, if not, the generalized scalar function provides the next entry (i.e. entry 11) to the column function. Specification, page 10, lines 1-4, Figure 4, item 164. This process repeats until the last entry (e.g. entry 14) is provided to the column function. Once the last entry is provided, then the column function completes its (normal and conventional) operations and provides the output. Specification, page 10, lines 4-6 and Figure 4, items 166 and 168. If other rows are also to be processed, then a similar method is carried out for additional rows. Specification, page 10, lines 6-8.

Thus, the methods (depicted in Figures 3-4), systems (depicted in Figure 5), and computer-readable media recited in claims 1-21 may provide row data entry by entry to the column function. Stated differently, the generalized scalar function simulates the column environment such that the data in the row can be provided to the column function as though the row were a column. The pre-existing and conventional column function operates normally. Specification, page 8, lines 17-19. As a result, the operations of conventional column function can be performed on row data. The conventional column function can, therefore, be used on

rows without rewriting the conventional column function. Specification, page 9, lines 1-4. The resources that would otherwise be expended on rewriting and testing the column function can be saved. Specification, page 8, lines 4-6.

VI. ISSUES

The issues presented are:

(1) whether the specification is adequate under 37 C.F.R. 1.71;

(2) whether claims 1-21 are each unpatentable under 35 U.S.C. § 112, first paragraph;

(3) whether claims 1-21 are each unpatentable under 35 U.S.C. § 112, second paragraph; and

(4) whether claims 1-21 are each unpatentable under 35 U.S.C. § 102 as being anticipated by U.S. Patent No. 6,289,336 (Melton).

VII. GROUPING OF CLAIMS

Appellant hereby states that claims 1-21 stand or fall together. Thus, claims 1-7, claims 8-14, and claims 15-21 are a single group having three independent claims. Therefore, Claims 1-21 constitute one (1) group.

VIII. ARGUMENTS

A. Summary of the Applied Rejections

In the above-identified Office Action, the Examiner objected to the specification under 37 C.F.R. 1.71 as failing to provide an adequate description of the invention. In so doing, the Examiner stated:

[i]t appears that applicants has [sic] attempted to incorporate some generalized scalar function and conventional column function into their specification to thereby support claims to any combination or permutation of features therefrom. The fact that features are mentioned individually does not mean that applicants have invented anything. There must be some evidence within the application filed that applicants were in possession of the claimed combinations. . . . Similarly, without more, the stand alone generalized scalar function and the conventional column function do not provide support for combining any features, regardless of what applicants may teach. . . To be effective in showing possession of the invention, an incorporation of the structures, links and mechanism derived from the generalized function to simulate the conventional column function environment which specifically contribute[s] to the claimed features and for what purposes should be disclosed.

The Examiner rejected claims 1-21 under 35 U.S.C. § 112, first paragraph, as containing subject matter which was not disclosed in the specification in such a manner as to allow one of ordinary skill in the art to make and/or use the invention. In so doing, the Examiner stated:

applicants fail to disclose the simulation mechanism and the corresponding data structures, for simulating the claimed scalar function with row parameter [sic] into [the] conventional column parameter environment. Particularly, the submitted specification fails to show the form of a generalized scalar function. It also miss[es] the structure being utilized to map the generalized function with a plurality of columns of a row into the claimed column function parameter. Furthermore, it fails to disclose the technique to initialize, evaluate and finalize the claimed column function.

The Examiner also rejected claims 1-21 under 35 U.S.C. § 112, second paragraph, as being indefinite for failing to point out and distinctly claim the subject matter that the Applicant regards as the invention. In so doing, the Examiner stated:

it is unknown what is the structure of the claimed generalized scalar function? How could this generalized function allow the at least one row to be provided to the claimed column function as though the at least one row was a column [i.e., a row may represent a record of a database table, which may comprise a plurality of columns. In this case, it is not understood how to map the plurality of columns into the claimed "a column"?].

Further, the Examiner rejected claims 1-21 under 35 U.S.C. § 102 as being anticipated by Melton. In so doing, the Examiner cited the "SQL compiler [118, Fig. 1; col. 3, line 65 – col. 4, line 11] to simulate a generalized scalar function [the Rows Since function, col. 2, lines 4-45] as the column function environment . . ."

Appellant respectfully requests that the Board reverse the Examiner's objection to the specification and the Examiner's final rejection of claims 1-21 under 35 U.S.C. § 112 first and second paragraphs and under 35 U.S.C. § 102.

B. The Cited Prior Art

Appellant agrees that Melton describes specific functions and a corresponding database query complier and its associated compilation methods. Melton, Abstract and Fig. 1. However, the cited portions of Melton describe a specific set of row functions that are written. In particular, cited portions of Melton describe row functions that are written to search previously accessed rows or offsets. Melton, col. 2, lines 10-45. As such, the functions in Melton are written to perform specific operations on rows. For example, the Rows Since function parses a search condition, identifies information from a previous row, and converts the information to provide an offset function based on the previously accessed row. Melton, col. 2, lines 10-19. The table is searched, looping through previous rows until the desire row, that satisfies the search condition, is found. Melton, col. 2, lines 20-23. The information in the row that satisfied the search condition is made available. Thus, information from a previous row can be accessed without having to change a cursor's position in a row. Melton, col. 2, lines 42-45.

C. The Specification is Adequate Under 37 C.F.R. 1. 171.

Appellant respectfully submits that the Examiner's objection to the specification under 36 C.F.R. 1.171 is without merit as the Examiner has completely failed to explain why the specification is insufficient to ensure that the inventor had possession, as of the filing date of the application, the specific subject matter claimed. In particular, The Examiner's objections appear

to rest on the use of the terms "generalized scalar function" and "column function." Appellant respectfully submits that the generalized scalar function and the column function are described in the specification with such particularity as to ensure that the Inventor had possession of the subject matter claimed.

As is clearly indicated in the specification, a column function is a preexisting, conventional function in relational databases. Such conventional column functions operate on a column of data and return a particular result. Specification, page 2, lines 11-13. For example, one example of a column function is a minimum function, which would return the minimum value in a particular column. Specification, page 2, lines 14-17. Other examples of conventional column functions given in the specification include maximum, sum and average, which return the maximum value, the sum of the values, and the average of the values, respectively, in the column(s) on which the conventional column functions operate. Specification, page 3, lines 8-12. In addition to providing well-known examples of column functions, in the Background of the Invention, the initialization phase, evaluation phase, and finalization phase of a conventional column function are also described. Specification, page 3, lines 13-19. Thus, given the fact that column functions are conventional preexisting and well known functions, the examples of conventional column functions provided in the specification, and the description of the phases of a conventional column function, Appellant respectfully submits that one of ordinary skill in the art would readily understand that the term column function. As such, Appellant respectfully submits that one of ordinary skill in the art would readily recognize how such conventional column functions operate.

Moreover, nothing new or different is done to the column function. See Declaration, paragraph 4. Instead, the embodiments of the method and system described in the specification

illustrate a new use for a conventional column function: row data. The generalized scalar function facilitates this new use as described in the specification and below. Appellant respectfully submits, therefore, that based on a reading of the specification, one of ordinary skill in the art would understand the term column function, would understand that the inventors have possession of the column function of the claimed invention, and would be capable of using the (pre-existing) column function in the manner recited in the claims and described in the specification.

The generalized scalar function is used to simulate a column environment for row(s) of data so that the row can be provided to the pre-existing column function as if the row were a column. Stated differently, the generalized scalar function simulates the column environment for the row(s) so that the column function can operate on the row data. Specification, page 8, lines 14-19. The specification also describes a specific embodiment of a generalized scalar function in which data from a row is provided entry-by-entry to the column function. Specification, page 9, line 14-page 10, line 8 and Figure 4. See also Paragraph 7 of the Declaration. Thus, in the embodiment described in the specification, the generalized scalar function fetches the row and individually provides the entries in the row to the column function. Consequently, the specification does describe one particular mechanism for simulating the column environment: providing the row data entry-by-entry to the column function. Figure 4, items 154, 156, and 164. This allows the column function to treat the entries in the row as though they are entries in a column. Figure 4, items 158, 160, 166, and 168. As a result, the resultant of the conventional column function can be obtained for row(s) in a table. Thus, both a generalized description of the function of one embodiment of the generalized scalar function (simulates the column environment for the row(s) so that the column function can operate on the row data) and the more-detailed description of one embodiment of the method, system, and computer-readable

medium in accordance with the present invention (provides the data entry-by-entry to the column function) are provided. In particular, one embodiment of a generalized scalar function and its communication with the column function are described in detail. See Figures 4 and 5. Consequently, Appellant respectfully submits that one of ordinary skill in the art would understand the term generalized scalar function and how such a generalized scalar function interfaces with a conventional column function. Appellant thus respectfully submits that one of ordinary skill in the art would understand the term generalized scalar function, would understand that the inventors have possession of the generalized scalar function of the claimed invention, and would be capable of using the generalized scalar function in the manner recited in the claims and described in the specification.

For the above-identified reasons Appellant respectfully requests that the Board reverse the Examiner's objection to the specification under 37 C.F.R. 1.171.

D. Claims 1-21 Are Not Unpatentable Under 35 U.S.C. § 112, First Paragraph.

Appellant respectfully submits that the applied rejections of claims 1-21 under 35 U.S.C. § 112, first paragraph, are without merit as the Examiner has completely failed to explain why the claims contain subject matter which was not disclosed in the specification in such a manner as to allow one of ordinary skill in the art to make and/or use the invention.

Independent claim 1 recites a method for using a "column function for a relational database in a structure query language (SQL) environment." Claim 1 also recites the use of a generalized scalar function, including the steps of allowing one or more rows of a table to specified as the argument for the generalized scalar function and using the generalized scalar function to simulate a column environment to "allow the at least one row to be provided to the column function as

though the at least one row was a column." Claim 1 also recites that the column function operates on the row(s) specified as arguments for the generalized scalar function and provides an output. Independent claims 8 and 15 recite analogous computer-readable medium and system claims. In the rejection in the above-identified Final Office Action, the Examiner appeared to focus on the terms column function, generalized scalar function, and a simulation mechanism for simulating the column environment. Consequently, these will primarily be discussed.

When read in light of the specification, the recited generalized scalar function, column function, and operations thereof are sufficiently clear to allow one of ordinary skill in the art to make and/or use the invention recited in varying scope in claims 1, 8, and 15. As discussed above, the specification describes both a column function and a generalized scalar function. In particular, the specification indicates that the column function is a conventional, preexisting function used in relational databases. In addition, the specification enumerates examples of conventional column functions such as minimum, maximum, sum, and average functions. The specification also indicates the phases that are typically found in such conventional column functions, namely the initialization, evaluation, and finalization phases. Thus, Appellant respectfully submits that one of ordinary skill in the art would recognize how to make and/or use a column function. See also, Declaration, paragraph 4.

The specification discusses the ability of a generalized scalar function to simulate, or mimic the column function environment. Thus, the mechanism for simulating a column environment is the generalized scalar function. As discussed above, the specification provides a detailed description of one embodiment of a method and system in accordance with the present invention. In particular, the embodiment describes selecting row(s), fetching row(s) and providing the data from the row(s) entry-by-entry to the column function. See also, Declaration, paragraphs 5, 7, and 8. Thus, both the

general function (simulate a column environment) and a specific embodiment (fetch row and provide data entry-by-entry to the column function) of a generalized scalar function are described. Furthermore, Appellant respectfully submits that the form of the generalized scalar function can be seen in Figure 4, which describes the functions performed including those performed by the generalized scalar function and the communication between the generalized scalar function and the column function, and Figure 5, which depicts the relationship between the column function, the generalized scalar function, and the user interface.

Thus, the column function, generalized scalar function, and the communication between the (conventional) column function are described in the specification with sufficient detail to allow one of ordinary skill to make and/or use the invention. The terms in independent claims 1, 8, and 15, such as generalized scalar function, simulation of the column environment, and column function are read in light of the specification. Consequently, Appellant respectfully submits that one of ordinary skill in the art would understand how to make and/or use the generalized scalar function in conjunction with the column function, as recited in claims 1, 8, and 15.

Claims 2-7, 9-14, and 16-21 depend upon independent claims 1, 8, and 15, respectively. Consequently, the arguments herein apply with full force to claims 2-7, 9-14, and 16-21. Appellant thus respectfully submits that one of ordinary skill in the art would be capable of making and/or using the subject matter recited in claims 1-21. Accordingly, Appellant respectfully requests that the Board reverse the Examiner's rejection of claims 1-21 under 35 U.S.C. § 112, first paragraph.

E. Claims 1-21 Are Not Unpatentable Under 35 U.S.C. § 112, Second Paragraph.

Appellant respectfully submits that the applied rejections of claims 1-21 under 35 U.S.C. § 112, second paragraph, are without merit as the Examiner has completely failed to explain why

claims 1-21 are indefinite for failing to point out and distinctly claim the subject matter the Appellant regards as the invention.

Independent claims 1, 8, and 15 recite the use of a generalized scalar function to simulate a column environment and a column function. When read in light of the specification, the generalized scalar function, the simulation of the column environment, and the column function are clear and definite.

As described in the specification, the column function is a conventional, column function such as the minimum, maximum, average, or sum functions. Specification, page 2, lines 11-17 and page 3, lines 8-12. Further, the column function operates in a normal, conventional manner. Specification, page 9, line 17-page 10, line 1 and Figure 4, items 158, 160, 166 and 168. Thus, the column function recited in claims 1, 8, and 15, and described in the specification is a conventional column function. Thus, recitation of the column function in claims 1, 8, and 15 would be clear and definite to one of ordinary skill in the art.

As discussed above, the generalized scalar function has a function that is recited and can be described at a high level as simulating the column environment such that the column function can operate on the entries of the row provided as an argument to the generalized scalar function. Specification, page 7, lines 21-23 and page 8, lines 10-13. In one embodiment, the simulation is carried out by providing the entries of the row entry-by-entry to the column function. Specification, page 9, lines 15-18 and page 10, lines 1-3. See also, Declaration, paragraph 7. Because the row data are provided entry-by-entry to the column function, the column function can operate on the row data as though the row(s) were column(s). Specification, page 8, lines 15-17 and Figures 4-5. Thus, at least one embodiment of the generalized scalar function is quite simple in nature—simply fetching a row and providing entries in the row entry-by-entry to the

column function. It is noted that each entry in the row corresponds to a different column. However, this is also described in the specification. See, specification, page 9, line 12-page 10, line 6 and Figure 1 and 4. Stated differently, the Examiner's mapping "a plurality of columns [i.e. the entries in a row which extends over multiple columns]" to a single column is, in one embodiment, carried out simply by providing the individual entries from the row (i.e. one entry from each column that the row extends across) to the column function. Thus, as described in the portions of the specification cited above, for row 6 of the table 1 depicted in Figure 1, one embodiment of the generalized scalar function would individually provide entries 11, 12, 13, and 14 to the appropriate column function. The column function would then operate in a conventional manner on these entries. Thus, when read in light of the specification, the term generalized scalar function, as well as simulation of the column environment, are clear and definite. Accordingly, Appellant respectfully requests that the Board reverse the Examiner's rejection of claims 1, 8, and 15 under 35 U.S.C. § 112, second paragraph.

Claims 2-7, 9-14, and 16-21 depend upon independent claims 1, 8, and 15, respectively. Consequently, the arguments herein apply with full force to claims 2-7, 9-14, and 16-21. Appellant thus respectfully submits that claims 1-21 are clear and definite. Accordingly, Appellant respectfully requests that the Board reverse the Examiner's rejection of claims 1-21 under 35 U.S.C. § 112, second paragraph.

F. Claims 1-21 Are Not Unpatentable Under 35 U.S.C. § 102.

Appellant respectfully submits that the applied rejections of claims 1-21 under 35 U.S.C. § 103 are without merit as the Examiner has completely failed to explain why Melton teaches or suggests the methods recited in claims 1-21.

Claim 1 recites a method including the steps of allowing a user to specify the at least one row as an argument for a generalized scalar function,

> simulating a column environment for the at least one row using the generalized scalar function to allow the at least one row to be provided to the column function as though the at least one row was a column; and . . . performing the column function on the at least one row to provide at least one output.

Thus, claim 1 recites that the generalized scalar function is used "to allow the at least one row to be provided to the column function as though the at least one row was a column." The column function can thus performed for data in the at least one row, which now mimics a column. Consequently, at least one output for the row data is provided by the column function. Claims 8 and 15 recite analogous computer-readable medium and system claims.

Thus, using the method, computer-readable medium and system recited in claims 1, 8, and 15, respectively, the pre-existing column function can be reused to work on row data. As a result, the resources that would be used in rewriting, testing, and debugging a row function that performs the operations of the column function are saved. Specification, page 9, lines 1-6; page 10, lines 9-13.

In contrast, the cited portions Melton fails to teach of suggest the use of the recited generalized scalar function in conjunction with a (pre-existing, conventional) column function. Instead, the cited portions of Melton describe a specific set of row functions that are written. In particular, the cited portions of Melton describe functions that are used to access rows without changing the position of the cursor within the table. These functions, therefore, perform specific operations, such as determining the offset between rows and search terms. Melton, col. 2, lines 10-45. Stated differently, the functions described in the cited portion of Melton are for rows and perform specific operations on these rows. Appellant can find no indication in the cited portions of Melton that the functions of Melton are used in conjunction with pre-existing column

functions. The cited portions of Melton are also devoid of mention of utilizing a generalized scalar function to simulate a column environment so that the row data appears to the column function as a column. For example, the cited portion of Melton also does not describe fetching a row and providing the row data to the column function as if the row was a client. More specifically, the cited portion of Melton does not describe an embodiment which provides the data from a row entry-by-entry to the conventional column function. Thus, the cited portions of Melton fails to teach or suggest using a generalized scalar function to allow the row(s) to be provided to the column function as though the at least one row was a column in conjunction with a column function that performs its operation in a conventional manner. Consequently, the cited portions of Melton fail to teach or suggest the method, computer-readable medium and system recited in claims 1, 8, and 15. Accordingly, Appellant respectfully submits that claims 1, 8, and 15 are allowable over the cited references and respectfully requests that the Board reverse the final rejection of claims 1-21 under 35 U.S.C. § 102.

Claims 2-7, 9-14, and 16-21 depend upon independent claims 1, 8, and 15, respectively. Consequently, claims 2-7, 9-14, and 16-21 are allowable for the same reasons discussed above with respect to claims 1, 8, and 15. Accordingly Appellant respectfully requests that the Board reverse the final rejection of claims 1-21 under 35 U.S.C. § 102.

G. Summary of Arguments

For all the foregoing reasons, it is respectfully submitted that the specification provides an adequate written description of the invention under 37 C.F.R. 1.71, that claims 1-21 (all the claims presently in the application) are patentable under 35 U.S.C. § 112 first and second paragraphs, and that claims 1-21 define subject matter which would not have been obvious under

35 U.S.C. § 103 or anticipated under 35 U.S.C. § 102 at the time the subject matter was invented. Thus, Appellant respectfully requests that the Board reverse the rejection of all the appealed claims and find each of these claims allowable.

Note: For convenience of detachment without disturbing the integrity of the remainder of pages of this Appeal Brief, Appellant's "APPENDIX" section is contained on separate sheets following the signatory portion of this Appeal Brief.

This Brief is being submitted in triplicate, and authorization for payment of the required Brief fee is contained in the transmittal letter for this Brief. Please charge any fee that may be necessary for the continued pendency of this application to Deposit Account No. 09-0460 (IBM Corporation).

Very truly yours,

June 29, 2004

Stephen G. Sullivan
Attorney for Appellant(s)
Reg. No. 30,801
(650) 493-4540

IX. APPENDIX

1. A method for utilizing a column function for a relational database in a structure query language (SQL) environment, the column function capable of performing an operation on an indeterminate number of entries, the relational database utilizing data including a plurality of entries being organized into at least one column and at least one row, the method comprising the steps of:

(a) allowing a user to specify the at least one row as an argument for a generalized scalar function;

(b) simulating a column environment for the at least one row using the generalized scalar function to allow the at least one row to be provided to the column function as though the at least one row was a column; and

(c) performing the column function on the at least one row to provide at least one output.

2. The method of claim 1 wherein the simulating step (b) further includes the steps of:

(b1) fetching a row of the at least one row; and

(b2) utilizing the generalized scalar function to provide the row to the column function as though the row was a column.

3. The method of claim 1 wherein the column function performing step (c) further includes the step of:

(c1) performing the column function on the row to provide an output; and wherein the method further includes the step of

(d) repeating steps (b1), (b2) and (c1) for each remaining row of the at least one row.

4. The method of claim 1 wherein the column function provides a maximum of each of the at least one row.

5. The method of claim 1 wherein the column function provides a minimum of each of the at least one row.

6. The method of claim 1 wherein the column function performing step (c) further includes the step of:

(c1) performing an initialization phase in response to a first entry of each of the at least one row;

(c2) performing an evaluation phase on each entry of the at least one row; and

(c3) performing a finalization phase after evaluation of a last entry of the at least one row.

7. The method of claim 1 wherein the generalized scalar function in combination with the column function allow the operation of the column function to be performed for the indeterminate number of entries in the at least one row.

8. A computer-readable medium containing a program for utilizing a column function for a relational database in a structure query language (SQL) environment, the column function capable of performing an operation on an indeterminate number of entries, the relational database utilizing data including a plurality of entries being organized into at least one column and at least one row, the program including instructions for:

(a) allowing a user to specify the at least one row as an argument for a generalized scalar function;

(b) simulating a column environment for the at least one row using the generalized scalar function to allow the at least one row to be provided to the column function as though the at least one row was a column; and

(c) performing the column function on the at least one row to provide at least one output.

9. The computer-readable medium of claim 8 wherein the simulating instructions (b) further includes instructions for:

(b1) fetching a row of the at least one row; and

(b2) utilizing the generalized scalar function to provide the row to the column function as though the row was a column.

10. The computer-readable medium of claim 8 wherein the column function performing instructions (c) further includes instructions for:

(c1) performing the column function on the row to provide an output; and wherein the program further includes instructions for

(d) repeating instructions (b1), (b2) and (c1) for each remaining row of the at least one row.

11. The computer-readable medium of claim 8 wherein the column function provides a maximum of each of the at least one row.

12. The computer-readable medium of claim 8 wherein the column function provides a minimum of each of the at least one row.

13. The computer-readable medium of claim 8 wherein the column function performing instruction (c) further includes instructions for:

(c1) performing an initialization phase in response to a first entry of each of the at least one row;

(c2) performing an evaluation phase on each entry of the at least one row; and

(c3) performing a finalization phase after evaluation of a last entry of the at least one row.

14. The computer readable medium of claim 8 wherein the generalized scalar function in combination with the column function allow the operation of the column function to be performed for the indeterminate number of entries in the at least one row.

15. A system for utilizing a column function for a relational database in a structure query language (SQL) environment, the relational database utilizing data including a plurality of entries being organized into at least one column and at least one row, the system comprising:

a column function capable of performing an operation on an indeterminate number of entries;

a generalized scalar function for simulating a column environment for the at least one row using the generalized scalar function to allow the at least one row to be provided to the column function as though the at least one row was a column such that the column function can perform an operation the at least one row to provide at least one output;

an interface for allowing a user to specify the at least one row as an argument for the generalized scalar function.

16. The system of claim 15 wherein the generalized scalar function further fetches a row of the at least one row and provides the row to the column function as though the row was a column.

17. The system of claim 15 wherein the column function further performs an operation on each of the at least one row to provide an output.

18. The system of claim 15 wherein the column function provides a maximum of each of the at least one row.

19. The system of claim 15 wherein the column function provides a minimum of each of the at least one row.

20. The system of claim 15 wherein the column function performs the operation by performing an initialization phase in response to a first entry of each of the at least one row, performing an evaluation phase on each entry of the at least one row and performing a finalization phase after evaluation of a last entry of the at least one row.

21. The system of claim 15 wherein the generalized scalar function in combination with the column function allow the operation of the column function to be performed for the indeterminate number of entries in the at least one row.